OCTERYX OS / FOUNDATION GUIDE 03

Action needs evidence.

An autonomous action is not accountable unless its source, context, authority, outcome and human boundary can be reconstructed.

FOUNDATION EDITION 0.1 / JULY 2026

MARK NICOLL - FOUNDER

Contents

A field guide to the governed context infrastructure beneath autonomous systems.

Read the canonical web edition and use the interactive working canvas.

01 / OPENING ARGUMENT

Every action creates an obligation to explain.

Capability tells us that a system can act. Evidence helps us understand whether it should have acted, what happened and what must change next.

When software recommends a task, an agent routes an instruction or a robot changes the physical environment, the result cannot be treated as a transient output. Someone may need to understand which event started the path, whose identity was resolved, what context was considered, which policy permitted the action, who retained authority and whether the outcome matched the request.

Logs are useful, but they are rarely organised around that question. They show what individual components emitted. They may omit rejected alternatives, the exact context supplied to a model, the policy version, the human confirmation or the relationship between a command and its physical receipt.

Evidence should be designed as part of the operating path. It is not an after-the-fact reporting feature. The system should preserve enough structured lineage to reconstruct material decisions while protecting sensitive context and keeping uncertainty visible.

> If the next reviewer cannot reconstruct the path, the environment has an output but not an accountable outcome.

02 / COMPLETION

A receipt is not an explanation.

"Task completed" describes a terminal state. It does not establish why the task was valid or what actually occurred.

A command can be accepted, rejected, cancelled, superseded, retried, partially completed or confirmed by a human. The physical effect may differ from the requested action. A robot can report success while the environment later shows that the intended outcome did not occur.

A durable receipt should therefore connect the command to the actor, target, capability, policy decision, idempotency key, execution state and observed outcome. It should not turn device confidence into unquestioned truth. Confirmation may need to come from an independent source or responsible human.

Question	Completion log	Accountable evidence
Requested	A task name or API call	Typed command, actor, target, purpose and constraints
Permitted	The endpoint returned success	Policy evaluation, authority and human boundary are linked
Performed	The device reported completion	Receipt and independent outcome evidence remain distinct
Corrected	The current status was overwritten	Supersession and correction lineage are retained

03 / DECISION LINEAGE

Preserve the chain, not just the answer.

Accountability depends on the handoffs between source, context, authority, action and outcome.

01	**Source**	Which event, request or observation started the path?
02	**Identity**	Which subject, actor, device and environment were resolved?
03	**Context**	Which facts, confidence, freshness and conflicts were materialised?
04	**Authority**	Which capability, consent, policy and human boundary applied?
05	**Action**	Which typed command was coordinated and with what constraints?
06	**Outcome**	What receipt, observation, review or correction followed?

Each link should refer to stable evidence rather than duplicating unrestricted sensitive data. The chain needs enough integrity to expose missing or substituted records, while access to the underlying context remains policy-controlled.

Evidence should travel at the boundary.

A reviewer should not need privileged access to every component in order to understand one material path.

An evidence bundle is a purpose-shaped collection of references, records and integrity information assembled for a defined review. It may support an operational handover, incident review, resident query, supplier discussion, assurance exercise or export.

The bundle should say what it covers, who requested it, which environment and subject scope apply, which redactions were made, which versions and keys were used and what remains unavailable. A valid signature can show integrity and origin; it cannot prove that a claim is true, a decision was lawful or an action was safe.

01	**Scope**	Environment, subject, event range and review purpose.
02	**Lineage**	References from source through decision and outcome.
03	**Versions**	Contracts, policy, context and canonicalisation in force.
04	**Integrity**	Checks, manifests and verification result at the boundary.
05	**Disclosure**	Classification, redactions and omitted material.
06	**Review**	Human owner, conclusion, correction and follow-up.

The most important action may be to stop.

A dependable system makes refusal, escalation and human confirmation observable outcomes.

Autonomy is often described as the ability to continue without interruption. In governed environments, restraint is part of capability. Unresolved identity, stale context, missing authority, conflicting instructions, consent ambiguity or unavailable evidence may require the system to withhold context, reject a command or pause for review.

The human pathway should identify who owns the decision, what evidence they receive, which alternatives remain available and how their confirmation or correction becomes part of the record. A generic "human in the loop" label is not enough.

In care, human care-decision ownership remains explicit. Robots, models and software can contribute signals, context and coordination without quietly acquiring clinical or care authority.

Proceed	**Permit within constraints**	Record the authority, conditions and responsible owner.
Clarify	**Request missing context**	Keep the original uncertainty and new evidence linked.
Escalate	**Route to the right human**	Preserve who received the case and why.
Stop	**Refuse or suspend action**	Treat safe non-action as a first-class outcome.

Evidence disappears at handoffs unless it is designed to cross them.

Every interface should state what it can attest, what it merely reports and what remains outside its authority.

Source system	**Origin, observation time and source record**	
Octeryx ingress	**Normalisation, scope and immutable event envelope**	
Identity and context	**Resolution, truth lane, freshness and materialisation**	
Policy	**Capability, consent, authority and evaluation record**	
Runtime	**Command, coordination, idempotency and receipt**	
Human reviewer	**Decision ownership, correction and conclusion**	
Export boundary	**Purpose, redaction, manifest and verification**	

A robot's onboard control and safety systems remain responsible for their own physical guarantees. A model remains a contributing component, not an authority for identity, consent, policy or truth. An application can present evidence without becoming the canonical operational record.

07 / CORRECTION

Correct the record without erasing the path.

Accountability requires a visible relationship between the original state, the correction and every downstream consequence.

Operational context changes. An identity match can be reversed. A source can retract a claim. A human can discover that an action receipt was incomplete. Sensitive material may require redaction from an export while remaining under stricter internal control.

Overwriting the original record makes later review impossible. Keeping every copy without access controls creates a different risk. The evidence design should append corrections and supersession links, preserve the reason and authority for the change, and apply policy to each representation.

Correction lineage allows the environment to learn without pretending that earlier decisions were made with information that did not yet exist.

The next actor should inherit a reconstructable handover.

A good handover carries changes, unresolved questions, owners and evidence rather than a flattened summary.

Humans, agents, applications and machines enter and leave an environment at different times. Each handover is a risk that context will be reduced to a sentence without sources, certainty or responsibility.

A reconstructable handover should distinguish what changed, what remains disputed, which action was taken, who owns the next decision, what must be revalidated and where supporting evidence can be accessed. The recipient should see the minimum necessary context, not the unrestricted history.

Shift	**Operational handover**	Changes, exceptions, owners, open actions and evidence references.
Machine	**Task handover**	Command state, constraints, stop conditions and receipt history.
Review	**Assurance handover**	Decision trace, versions, redactions, correction and conclusion.
Resident	**Human explanation**	Plain-language purpose, action, responsible owner and next step.

09 / WORKING TOOL

Decision reconstruction canvas.

Use this after a proposed or completed action to find the first point where explanation, authority or outcome becomes unverifiable.

PRINTABLE DECISION RECONSTRUCTION CANVAS

Record the status, owner and evidence for each link in the decision path. The first unowned link is the most urgent reconstruction gap.

AREA	RECONSTRUCTION QUESTION	STATUS / OWNER / EVIDENCE
Trigger	Can the originating event, request or observation be identified?	
Identity	Can the subject, actor, device and environment resolution be reviewed?	
Context	Is the exact materialised context, including freshness and conflict, recoverable?	
Authority	Are capability, consent, policy and human responsibility linked?	
Decision	Are considered options, stop conditions and selected path visible?	
Receipt	Can acceptance, execution state, retries and idempotency be reviewed?	
Outcome	Is the observed effect distinct from the device or application report?	
Correction	Can review, redaction, supersession and follow-up be traced?	

The canvas does not prove that an action was correct. It tests whether the path can be examined. A complete reconstruction may still expose a poor decision; an incomplete one prevents meaningful review.

An evidence architecture is not operational assurance.

Octeryx OS is in active R&D. This guide describes intended evidence principles and accepted target architecture.

Currently evidenced

- Accepted modular product doctrine, architecture and implementation plan
- Typed event, provenance, policy, decision, command, receipt and evidence contracts
- Deterministic fixtures, repository validation and explicit claim controls
- Target PostgreSQL, Kasa and Film ownership boundaries

Not currently claimed

- Live production or care deployment
- Clinical validation, certification or regulatory approval
- Production readiness or universal robot compatibility
- Operational cryptographic assurance or post-quantum protection
- Unsupervised autonomous care decisions or robot operation

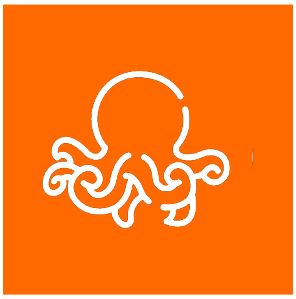
Contracts and deterministic tests can show that the intended evidence vocabulary is precise and reviewable. They do not establish production safety, certified operation, regulatory compliance, live key custody or real-world outcomes. Read current statements with the canonical facts and proof boundary.

Octeryx OS is building context that can explain its path.

One context operating system, with Kasa as the primary care profile and Film as one isolated adjacent profile.

Category	Context operating system for safe, explainable physical-world autonomy
Care framing	Resident-context operating layer for autonomous care
Evidence thesis	Material actions should link source, context, authority, receipt and outcome
Current stage	Active R&D / Accepted architecture and deterministic repository proof
Human boundary	Human care-decision ownership remains explicit

This guide is educational material, not a clinical, safety, engineering, procurement, regulatory, legal or investment recommendation. Product and maturity statements should be read with the current Octeryx OS proof boundary.



OCTERYX OS

Action Needs Evidence

Octeryx OS is building a context operating system for safe, explainable physical-world autonomy, with Kasa as the primary care profile and Film as one isolated adjacent profile.

octeryx.com

MARK NICOLL - FOUNDER

Educational material only. This publication is not a clinical, safety, engineering, procurement, regulatory, legal or investment recommendation. Product and maturity statements should be read with the current Octeryx OS proof boundary.